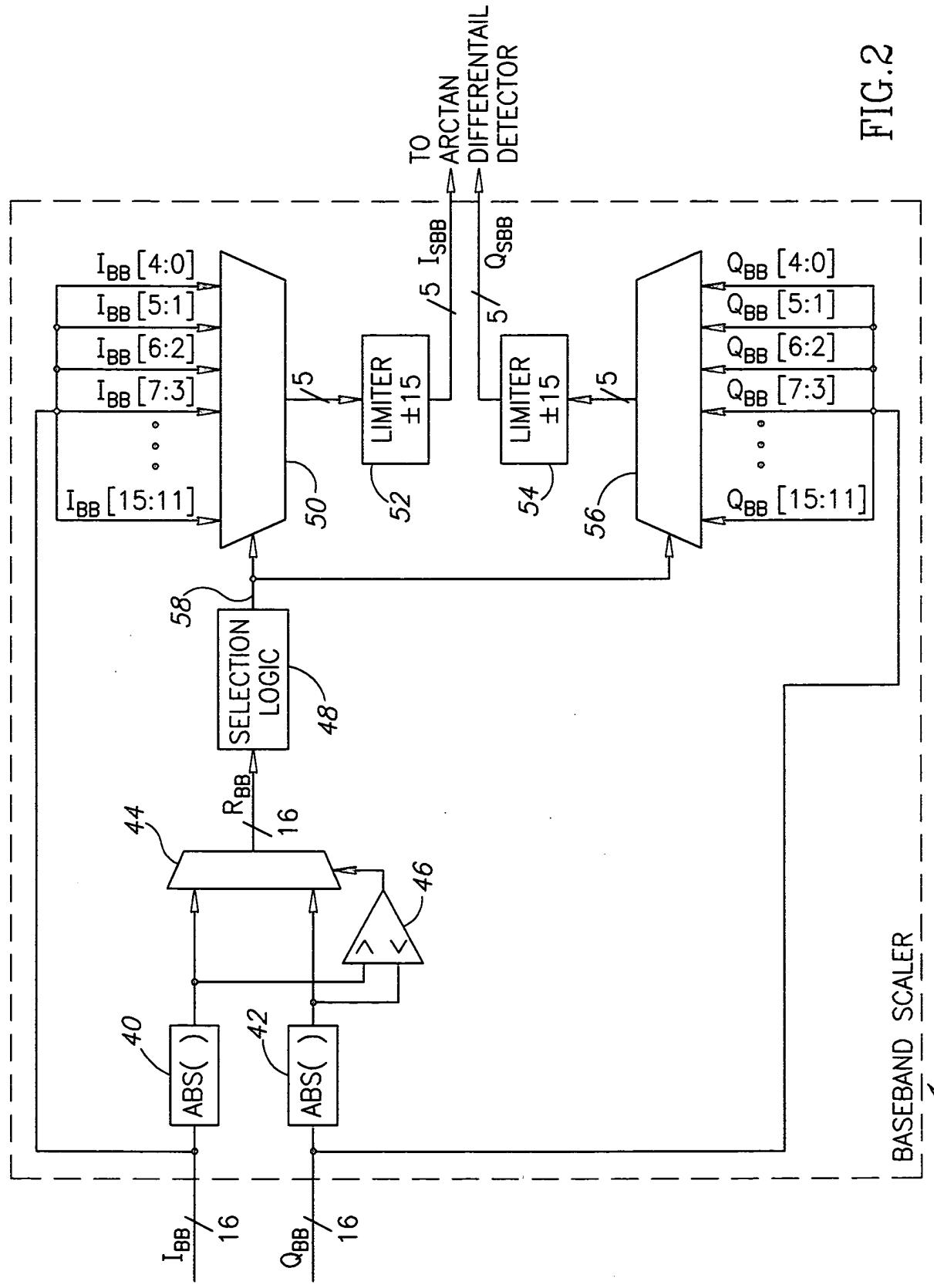


FIG.1

FIG.2



## LISTING 1

```
if (RBB [14]=1)
    ISBB = IBB [15:11];
    QSBB = QBB [15:11];

elseif (RBB [13]=1)
    ISBB = IBB [14:10];
    QSBB = QBB [14:10];

    .
    .
    .
    .

elseif (RBB [4]=1)
    ISBB = IBB [5:1];
    QSBB = QBB [5:1];

else
    ISBB = IBB [4:0];
    QSBB = QBB [4:0];

end
```

FIG.3

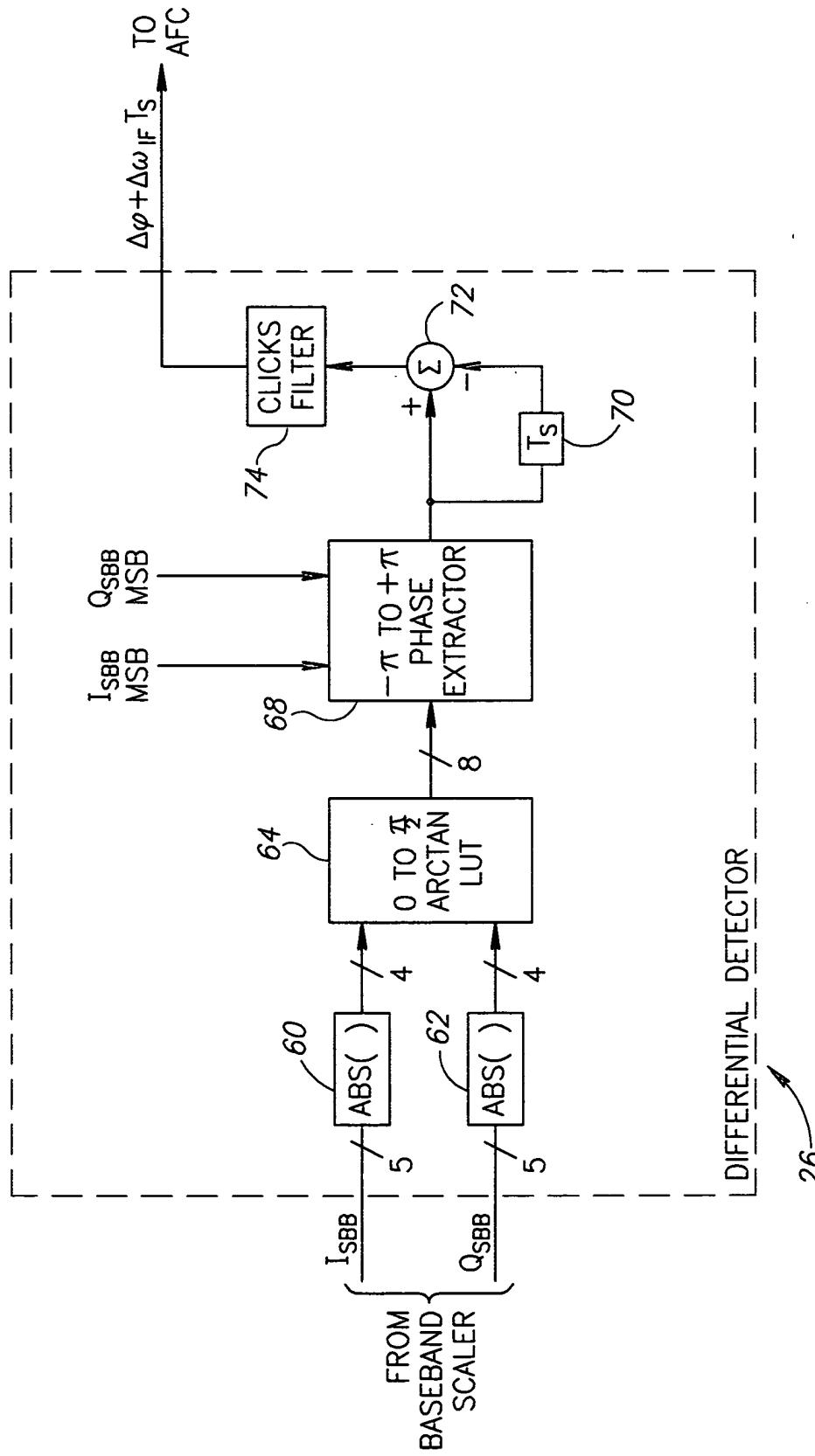


FIG. 4

ATAN LUT CONTENTS		$\varphi$ PHASE OUTPUT														
Q <sub>SB</sub>	I <sub>SB</sub>	101	59	41	31	25	21	18	16	14	13	12	11	10	9	9
140	101	75	59	49	41	36	31	28	25	23	21	20	18	17		
160	126	101	82	69	59	52	46	41	37	34	31	29	27	25		
170	142	119	101	86	75	66	59	54	49	45	41	38	36	33		
176	152	132	115	101	89	79	72	65	59	55	51	47	44	41		
180	160	142	126	112	101	91	82	75	69	64	59	55	52	49		
183	165	149	135	122	110	101	92	85	78	73	68	63	59	56		
185	170	155	142	130	119	109	101	93	86	80	75	71	66	63		
187	173	160	148	136	126	116	108	101	94	88	82	78	73	69		
188	176	164	152	142	132	123	115	107	101	94	89	84	79	75		
189	178	167	156	146	137	129	121	113	107	101	95	90	85	81		
190	180	170	160	151	142	133	126	119	112	106	101	95	91	86		
191	182	172	163	154	146	138	130	124	117	111	106	101	96	91		
192	183	174	165	157	149	142	135	128	122	116	110	105	101	96		
193	184	176	168	160	152	145	138	132	126	120	115	110	105	101		

FIG.5

## LISTING 2

```
if QSBB=0 then
    arctan LUT component output is 0;
elseif ISBB=0 then
    arctan LUT component output is 201 (=round(128*π/2));
else
    arctan LUT component output is according to the matrix
    of Fig.5, where the upper left corner is the element of
    (1,1) indexes and the lower right corner is of (15,15)
    indexes;
end
```

FIG.6

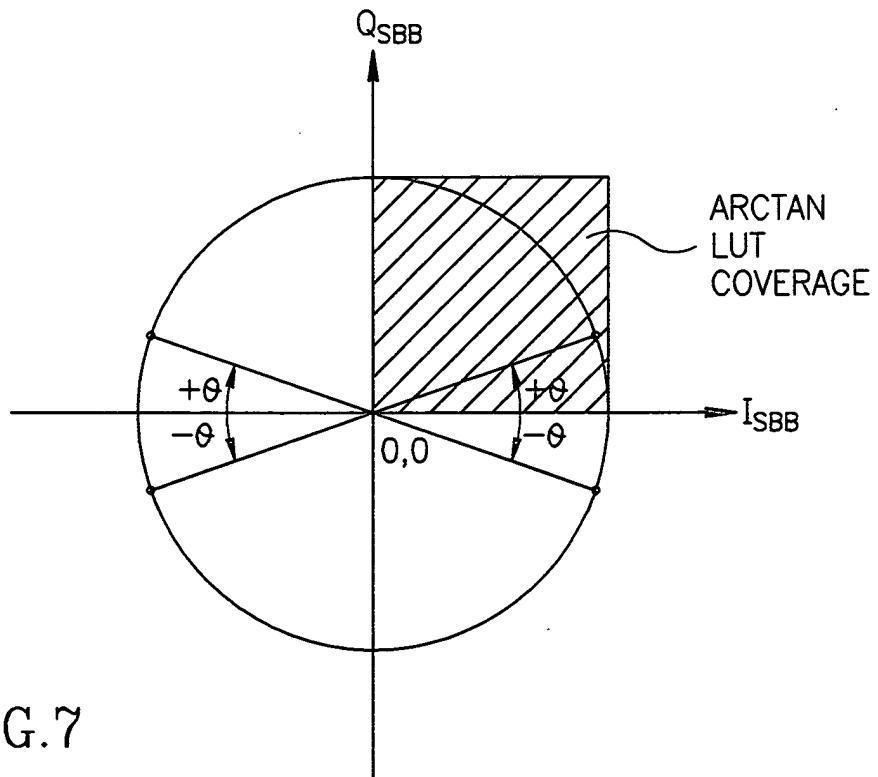


FIG.7

### LISTING 3

```
if (sign(ISBB(k)) >= 0
    if (sign(QSBB(k)) >= 0
        phi(k) = phi_base(k);
    else
        phi(k) = -phi_base(k);
    end
else
    if (sign(QSBB(k)) >= 0)
        phi(k) = round( $\pi$ +128)-phi_base(k);
    else
        phi(k) = phi_base(k)-round( $\pi$ +128);
    end
end
```

FIG.8

### LISTING 4

```
if (delta_phi(k) > round( $\pi$ ))
    delta_phi(k) = delta_phi(k)-round(2 $\pi$ );
elseif (delta_phi(k) < -round( $\pi$ )
    delta_phi(k) = delta_phi(k)+round(2 $\pi$ );
```

FIG.9